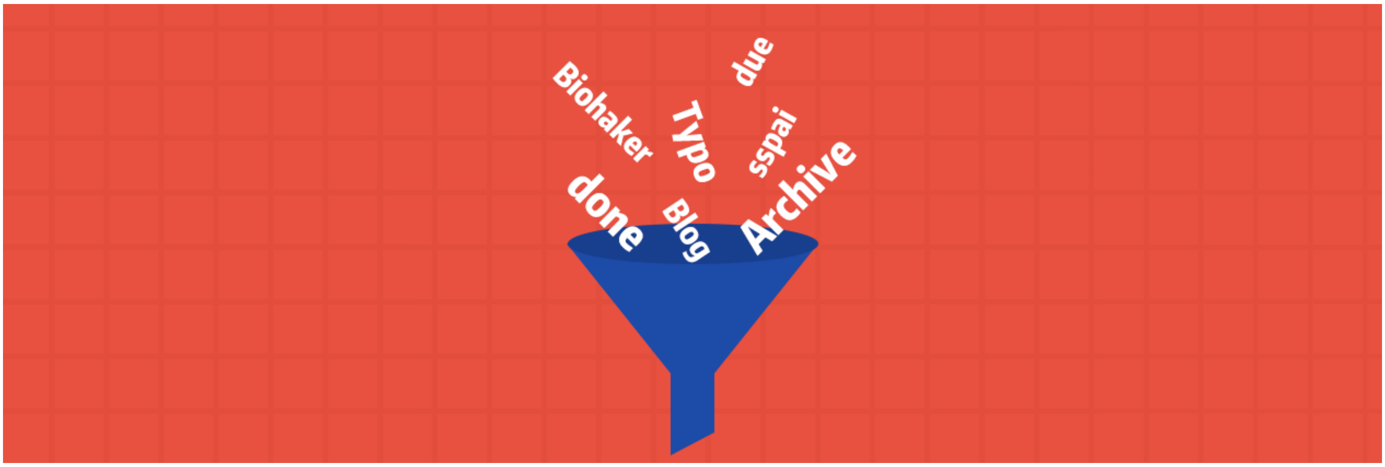
【08】第三章 | 定位和过滤



当你建立了一个事无巨细的列表，随之而来的问题就是怎么找到你需要的东西。

定位和过滤是两种「寻找」的方法，其本质区别就在于一个是快速找到任务所在位置，一个则是把任务汇集在你眼前。列表从宏观入手，标签自微观切入。

和多数任务管理软件一样，TaskPaper 里的列表（项目）主要用于定位，标签则司过滤。但是它也有一个独特的设计：sidebar 和 Go To Anything，里面汇集了所有的标签（除了个别特意排除掉的）、列表和搜索动作。对于多数不复杂的情景，你可以不用特意去想到底该用哪种搜索方式，只要输入模糊的关键词。

> The first to know about search in TaskPaper is that you never need to think about them.（关于 TaskPaper 里的搜索，首先要知道的就是你不必想着要搜什么）——Jesse（TaskPaper 开发者）

Sidebar 和 Go To Anything 两个功能实质上是一样的，但是前者提供了更倾向光标点选的操作，后者则通过打字的方式检索，近似 Spotlight 和 Alfred。

Go To Anything 自动匹配

Sidebar 其实也有自动匹配操作：当你用光标点一下该区域，然后任意按一个字母，就能定位到对应首字母的项，例如按一下 T 选中「Today」；没有该首字母的话就匹配第二个字母，依此类推；遇到字母冲突的时候，像是「Today」和「TaskPaper」都由「T」打头，则可以通过先后（间隙极短）按下 T 和 O 定位到「Today」搜索。

Sidebar 自动匹配

无论是 sidebar 还是 Go To Anything，包括之后会详细介绍的标签组，它们的自动匹配都只支持英文。我建议项目名、标签（组）名以及搜索动作名——总之 sidebar 里的一切，都使用英文，便于进行自动匹配。

Sideabar 在一定程度上模糊了标签、项目和搜索之间的区别，让你找任务更轻松，体验类似文本编辑器内的⌘Command+F。至于更精确的定位与搜索，则可以下文的方法实现。

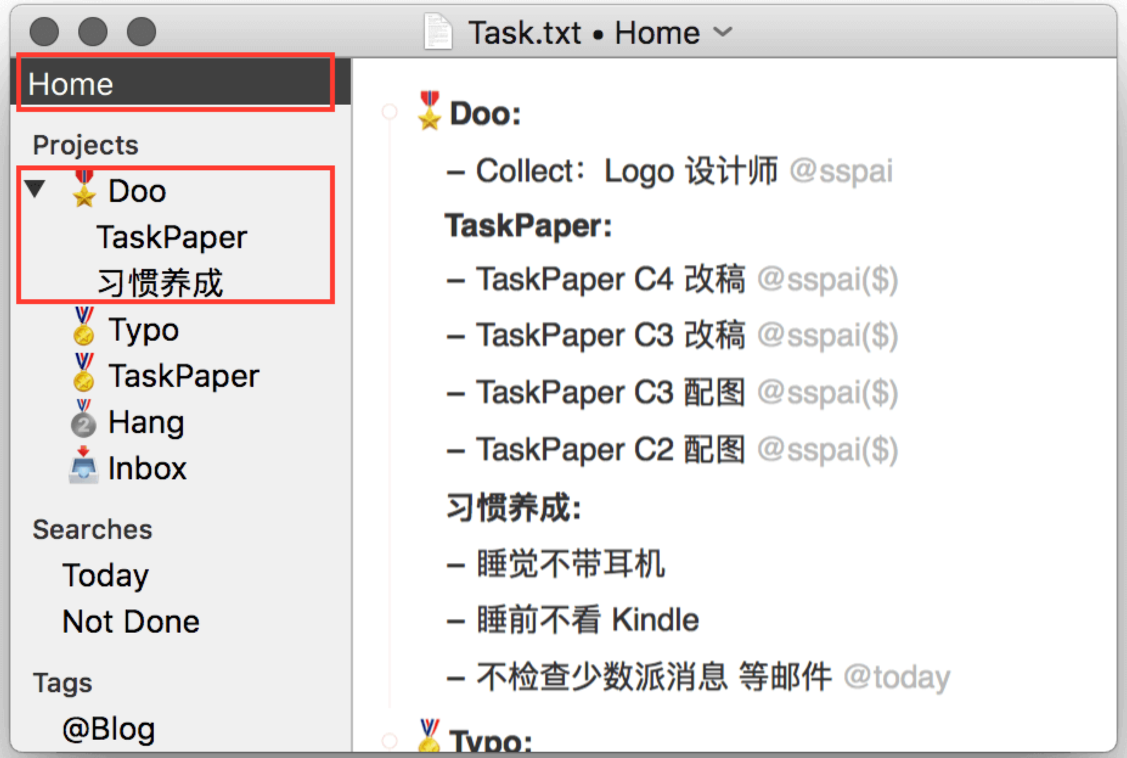
通过列表快速定位

列表在 TaskPaper 相当于文件夹，与标签不同，任务和列表是函数一样一一对应的关系，任务好像菜单上的菜品，有着自己固定的位置，这就为定位提供了方便。

主列表、项目和任务组

TaskPaper 支持无限层级，如果你需要，可以项目里嵌套项目、项目任务互相嵌套，但基于列表的目的——**结构化文本，以便检视**、**快速定位**——劝君惜墨如金，够用即可，我们能消受得起的不过三四层，再复杂下去反而不利于查找定位——就像普通用户看到 Linux 命令行内太长的路径会头疼。

项目之下可以设置子项目，同样任务组之下也能继续分层设组，不过它们在 TaskPaper 里都视为一种列表。唯一的区别是，项目和子项目存在于 sidebar 和 Go To Anything，「曝光率」更高，而任务组则无此殊荣——如果一系列事情特别重要，我就会把它们组织在项目下面，便于快速找到这些任务。近期我想调整自己的作息，就在每天需要完成的 Doo: 项目里特意加了一组 习惯养成: 子项目。


Task.txt • Home
Home
Projects
Doo
TaskPaper
习惯养成
Typo
TaskPaper
Hang
Inbox
Searches
Today
Not Done
Tags
@Blog

Doo:

- Collect：Logo 设计师 @sspai

TaskPaper:

- TaskPaper C4 改稿 @sspai($)
- TaskPaper C3 改稿 @sspai($)
- TaskPaper C3 配图 @sspai($)
- TaskPaper C2 配图 @sspai($)

习惯养成:

- 睡觉不带耳机
- 睡前不看 Kindle
- 不检查少数派消息 等邮件 @today

Typo:

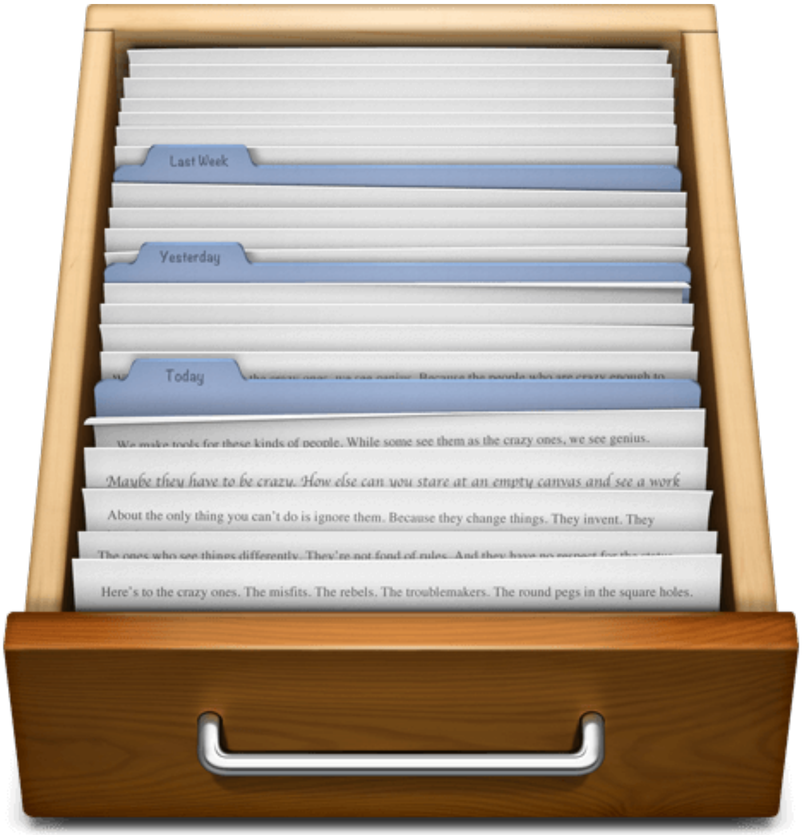
主列表和子项目

此外，sidebar 里面还有一个特殊的列表：主列表（Home），一份 taskpaper 文档里的所有任务都陈列于此。点击上图左上角位置就进入主列表，我会每天简单扫一遍，偶尔能在搁置任务中发现一些灵感——这篇文章改稿前，我就揪出了一个闲置已久的任务：- 干掉系统垃圾文件 @写作环境，刚好昨天在一本工具书上看到相似例子，旋即写了一个脚本解决问题。

建议读者们也不妨在主列表里俯瞰整个任务安排，站在这个更高的视角，也许能做出更棒的规划。

起初我带入了其他工具对各种列表的复杂定义，但随后发现在 TaskPaper 里不用过度解读，毕竟它只是个文本文档，只需依照符合自己的直觉的方式创建列表：追求简单直观。

TaskPaper 默认的样式就不错。一份 taskpaper 文件里的所有内容都呈现在 Home 列表，里面包含了列表，通过缩进来分隔它们；对于项目，还会采用加粗字体的项目名，整个 Home 列表非常直观。其实不论你点进哪个列表，里面都是 TaskPaper 四大基本元素组成的， **切换到任何一处都保持了观感上的干净和统一。**


Last Week
Yesterday
Today
We make tools for these kinds of people. While some see them as the crazy ones, we see genius.
Maybe they have to be crazy. How else can you stare at an empty canvas and see a work
About the only thing you can't do is ignore them. Because they change things. They invent. They
Here's to the crazy ones. The misfits. The rebels. The troublemakers. The round pegs in the square holes.

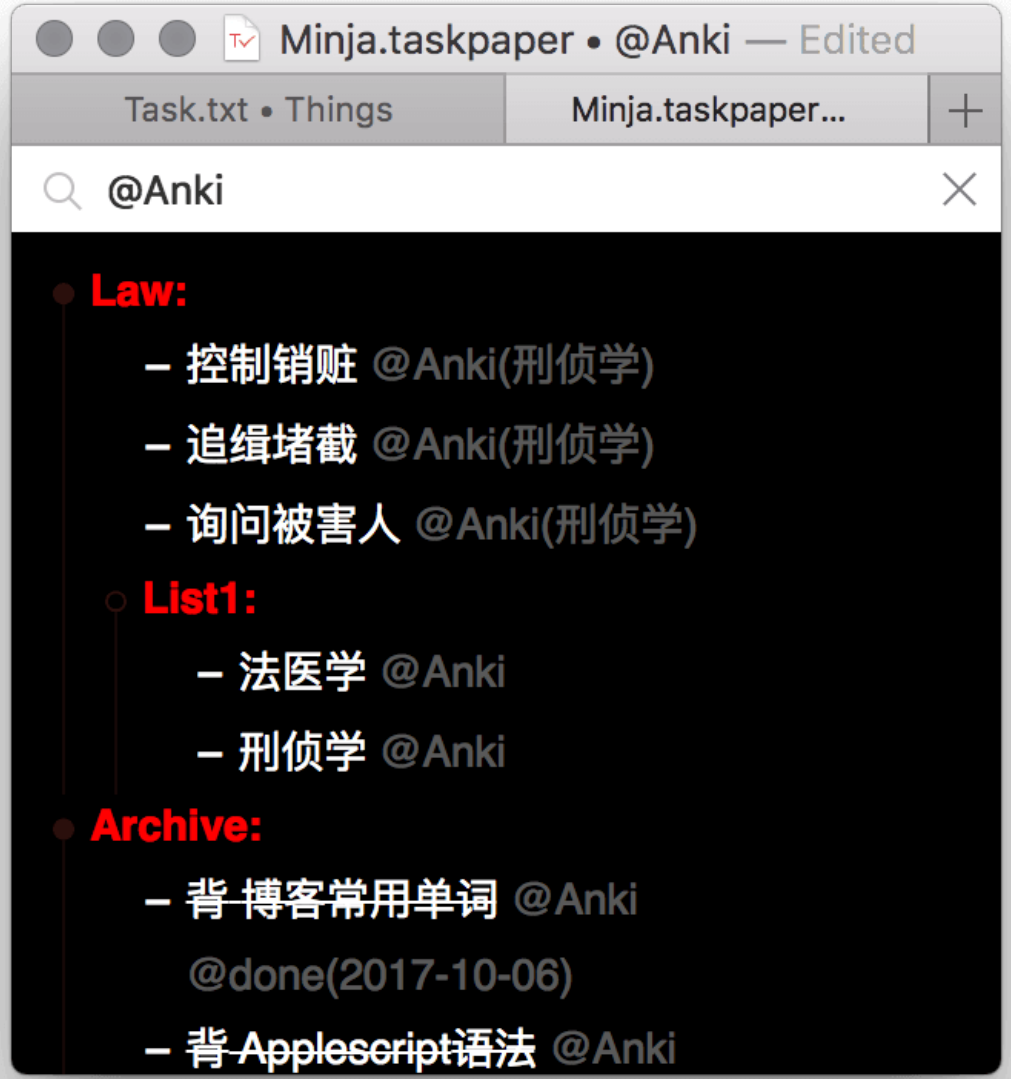
列表和案卷

项目和任务的显示方式，非常像物理世界中的档案抽屉。在 TaskPaper 的设计理念中，无论这一任务管理工具增加了多少功能，始终不脱离列表这一本质。它最大限度地摒弃图形

界面，降低用户学习、理解列表组织方式的成本，**依靠 sidebar 和列表本身的形状来定位。**

给列表「上色」

如果你觉得，在视觉上默认的灰白配色还不能很好起到分隔作用，可以通过 StyleSheet 自定义色彩。在 Finder 中使用快捷键 ⌘Command+⇧Shift+G，粘贴文件夹路径 ~/Library/Application Support/TaskPaper/StyleSheets/ 并前往，在该目录下可见默认的配色文件「Light.less」。我原地复制了一份，改名换姓为 Dark，方便在多个主题中来回切换。TaskPaper 使用 LESS（一门 CSS 预处理语言）文件来配色（以及其他视觉效果）。这里举一个非常简单的例子，我特意把项目名设置为显眼的红色：


Minja.taskpaper • @Anki — Edited
Task.txt • Things
Minja.taskpaper...
@Anki

Law:

- 控制销赃 @Anki(刑侦学)
- 追缉堵截 @Anki(刑侦学)
- 询问被害人 @Anki(刑侦学)

List1:

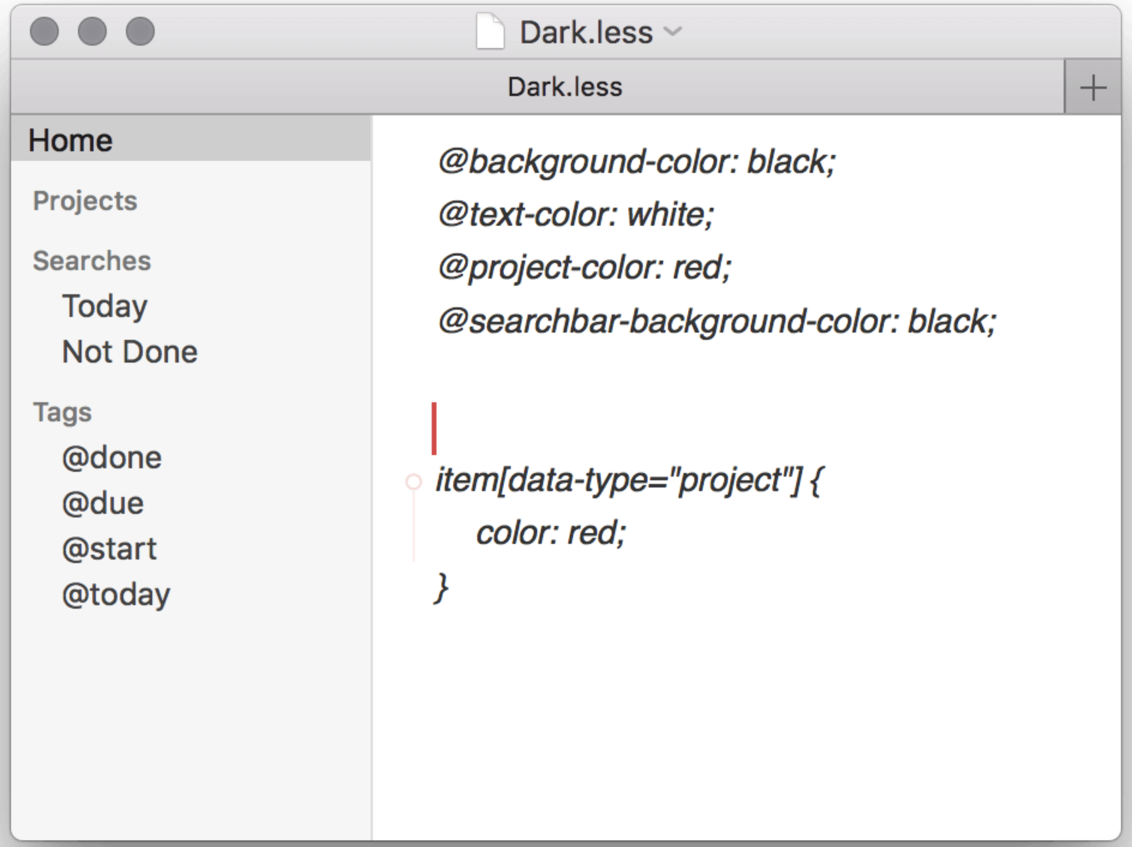
- 法医学 @Anki
- 刑侦学 @Anki

Archive:

- ~~背 博客常用单词~~ @Anki @done(2017-10-06)
- ~~背 Applescript语法~~ @Anki

配色效果

不需要其他任何应用，使用 TaskPaper 就可以直接打开并编辑 LESS 文件，就像编辑一份任务文件一样，一行一行往里面加配色语句就可以。


Dark.less
Dark.less
Home
Projects
Searches
Today
Not Done
Tags
@done
@due
@start
@today

```
@background-color: black;
@text-color: white;
@project-color: red;
@searchbar-background-color: black;


item[data-type="project"] {
    color: red;
}
```

配色编辑

官方提供了非常丰富的可选项，我挑了对于定位最实用的几项。当然可以自定义的远远不止颜色，这里只是最简单的一次运用。

语法	作用
@background-color: 颜色;	编辑器背景颜色
@text-color: 颜色;	文字默认颜色
item[data-type=“project”] {color: 颜色;}	项目颜色
run[tag=“data-标签名”] {color: 颜色;}	标签颜色

颜色可以用简单的小写英文单词，red、blue 这些常见单词可以被识别，**在冒号和颜色之间需要一个英文空格隔开**；需要更复杂的颜色，请用 rgb 或 hsl 格式的颜色表达式，例如：

rgb(98%,98%,98%)

hsl(187, 47%, 55%)

上面例子中的颜色是比较简单的，但可能才华横溢的你为了华丽的配色效果，写了非常冗长的配置语句，那也可以像这样分分行，以便自己阅读：

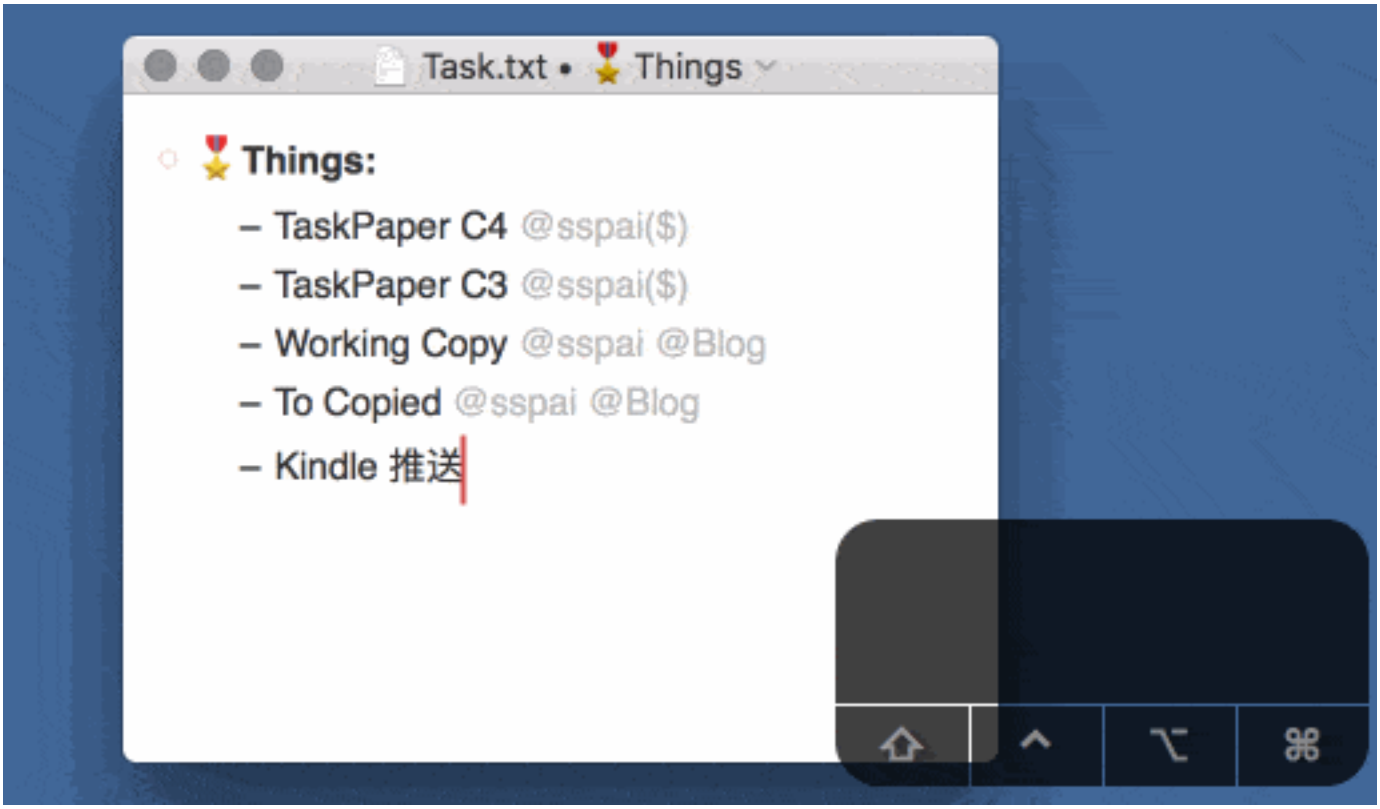
写好后保存一下，在 TaskPaper 菜单栏的 Window-StyleSheet 里选择刚才的 Dark.less，就看到配色起作用了。

相信你已经学会了简单的配色方法。如果你一时间没有什么色彩搭配的头绪，这里也有一份我写好的配置文件，可以试一试：点此下载

有趣的是，TaskPaper 其他配置文件同样可以打开为整齐划一的列表，如果你常常用 TaskPaper 来配置 TaskPaper，会有一种微妙的感觉；不借助其他工具就可以配置自身（体验和记录任务完全一样），这也是 TaskPaper 的简单、统一之处。对于普通用户，StyleSheet 他可以不用，但是某一天心血来潮或意识到需求之时，可以没太大台阶就上手。

层级

各种列表依靠层级来组织，所谓同一层就是一行内容前的缩进一样多——看起来靠左对齐；任务或项目越靠左层级越高。修改层级在 TaskPaper 里很简单。我们可以简单地用 ⇥Tab 和 ⇧Shift+⇥Tab 将某一行往后/往前移动一层，也可以用 ^Control+⌘Command+方向键 把它往前后、上下移动。



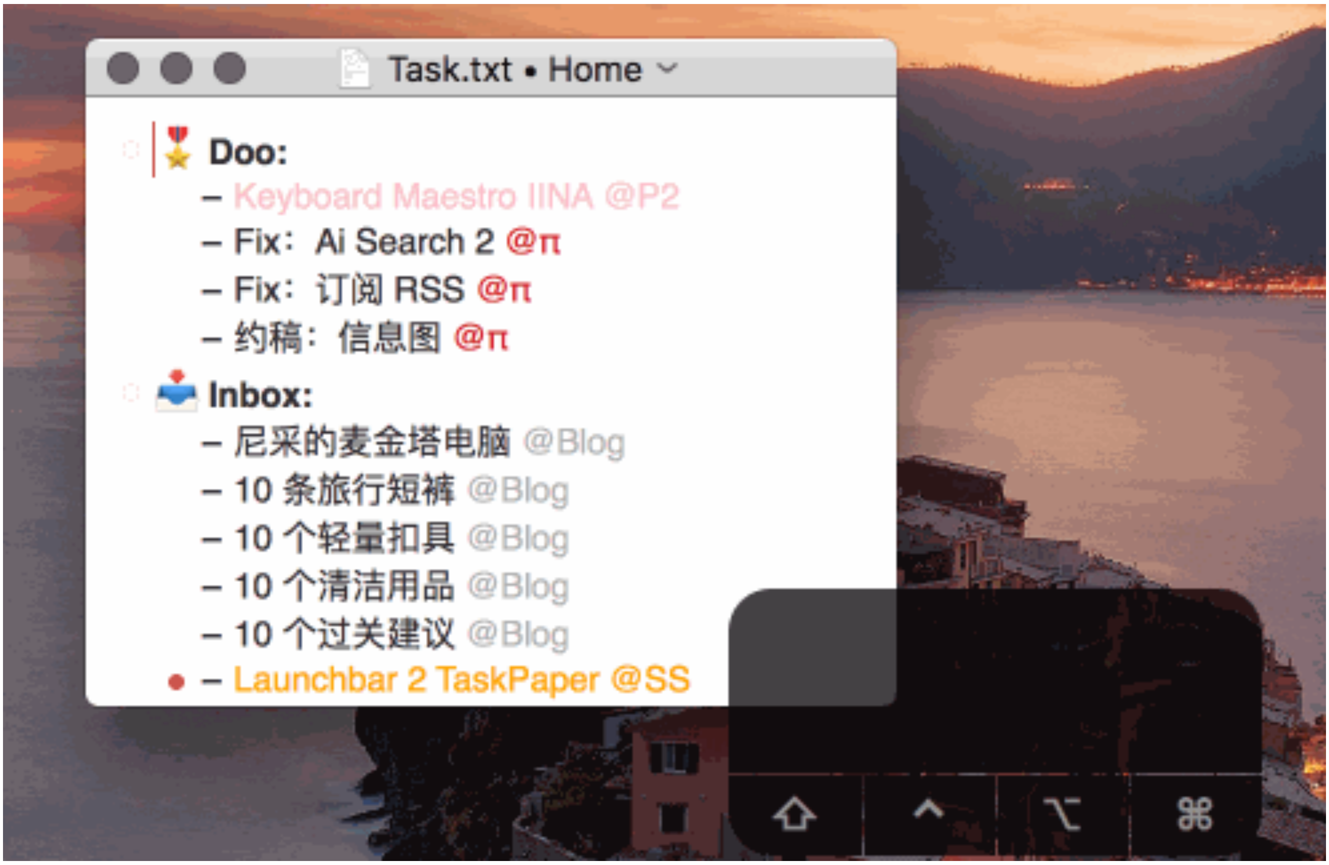
🎖 Things:
- TaskPaper C4 @sspai($)
- TaskPaper C3 @sspai($)
- Working Copy @sspai @Blog
- To Copied @sspai @Blog
- Kindle 推送

层级移动

批量操作也是一样的快捷键，如果没有批量选中，层级操作只会针对当前行，改变项目的层级并不会影响下面的任务。但你把某一个项目或任务组折叠起来的话，移动又是整组进行的了。

聚焦

除了直接在 sidebar 或 Go To Anything 里定位，你也可以用聚焦快捷键展开显示某个项目或任务组。所谓聚焦，是基于列表的一种定位方式，通俗讲就是只看某一个列表，把其他内容都隐藏起来。Sidebar 和 Go To Anything 像是目录，可以跳转到对应列表；而使用快捷键，则是像放大镜一样，展开当前的列表、排除其他的列表。



Task.txt • Home

Doo:
- Keyboard Maestro IINA @P2
- Fix：Ai Search 2 @π
- Fix：订阅 RSS @π
- 约稿：信息图 @π

Inbox:
- 尼采的麦金塔电脑 @Blog
- 10 条旅行短裤 @Blog
- 10 个轻量扣具 @Blog
- 10 个清洁用品 @Blog
- 10 个过关建议 @Blog
- Launchbar 2 TaskPaper @SS

聚焦

聚焦的快捷键是 ⌥Option+⌘Command+←/→。当你扫视整个列表的时候，恰好遇到某个想仔细看看的项目，用 ⌥option+⌘command+→ 聚焦到它身上准没错。

标签和搜索

标签并不是一个高深的概念，我们早在不知道什么是任务管

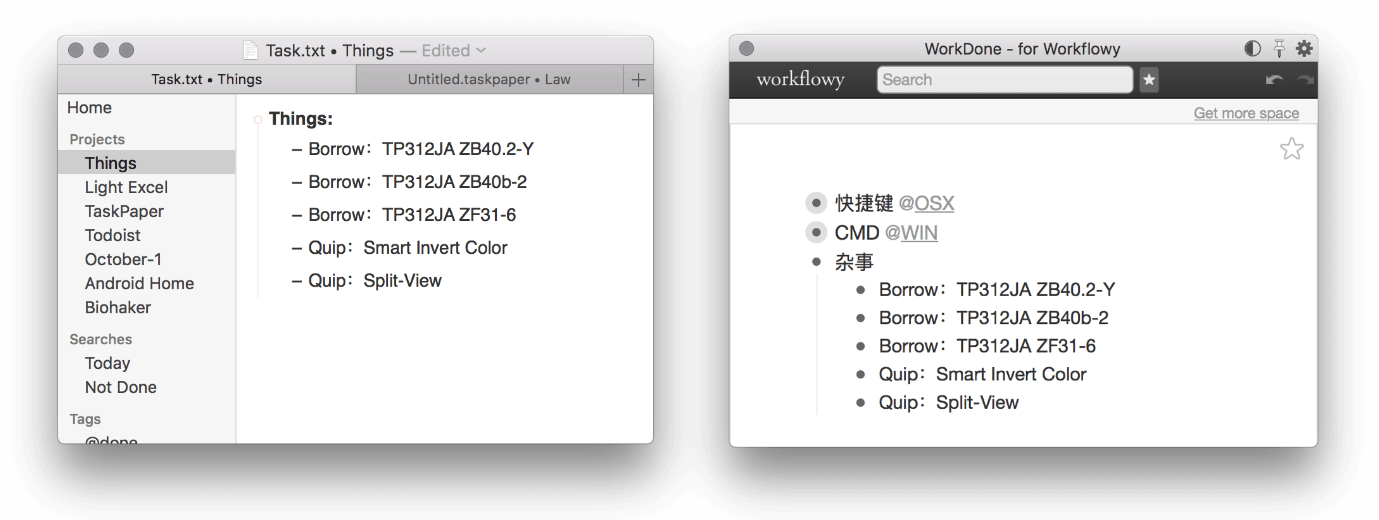
理的时候，就已经在使用它们了。笔记簿里的书签、大部分参考书侧面的索引，都是一种标签。

但是物理世界中的标签作用有限，只能指示一个特定的位置，而智能设备上的应用则把标签提升了一个等级，获得了过滤的作用。我们不仅仅满足于知道任务在哪，更需要的是把它们汇集在眼前，这是纸笔无法满足的——即使是灵活的活页本 + 便利贴组合，也是以「张」为单位存储笔记，应用则可以做到字符级别。

使用 TaskPaper 的初期，我时常回想起 Workflowy——一个广受欢迎的清单工具，许多人拿它来做任务管理。

TaskPaper 和 Workflowy 乍一看是非常相似的，但之间的区别，可以很好地说明 TaskPaper 作为高级任务管理工具究竟优势在哪：

- TaskPaper 基于本地的文档，Workflowy 非常依赖云端的数据；
- TaskPaper 拥有完备的过滤语法，Workflowy 仅能查一个关键词。


Task.txt • Things — Edited
Task.txt • Things
Untitled.taskpaper • Law
Home
Projects
Things
Light Excel
TaskPaper
Todoist
October-1
Android Home
Biohaker
Searches
Today
Not Done
Tags
WorkDone - for Workflowy
workflowy
Search
Get more space

Things:
- Borrow：TP312JA ZB40.2-Y
- Borrow：TP312JA ZB40b-2
- Borrow：TP312JA ZF31-6
- Quip：Smart Invert Color
- Quip：Split-View

- 快捷键 @OSX
- CMD @WIN
- 杂事
  - Borrow：TP312JA ZB40.2-Y
  - Borrow：TP312JA ZB40b-2
  - Borrow：TP312JA ZF31-6
  - Quip：Smart Invert Color
  - Quip：Split-View

TaskPaper 和 Workflowy

关于第一点，我会在下一章集中介绍文档文件的优点，目前你只需知道它要灵活的多；但第二点，即过滤，则是包括 Workflowy 在内众多应用无法逾越的高峰（也是大纲（Outliner）工具和真正的任务管理工具间的分水岭）。

标签组

TaskPaper 的搜索之所以强大，在于其合理的标签系统，使用搜索离不开标签和标签组。这个「高级货」的公式其实非常简练：

这就引入了标签组的概念（有编程经验的读者可以用「类」来理解）。TaskPaper 非常依赖标签进行搜索，用的时间长了，容易积累大量标签（包括了意义重叠、相近的标签），此时你可能绞尽脑汁回忆标签是什么——这就本末倒置。TaskPaper 的解决方案是用标签组来组织标签，把相关的一类标签统一起来管理。当然，标签组不是必须的，像 TaskPaper 预设的 @done 默认就没有设值。在你需要的时候设置它，而不必追求形式。

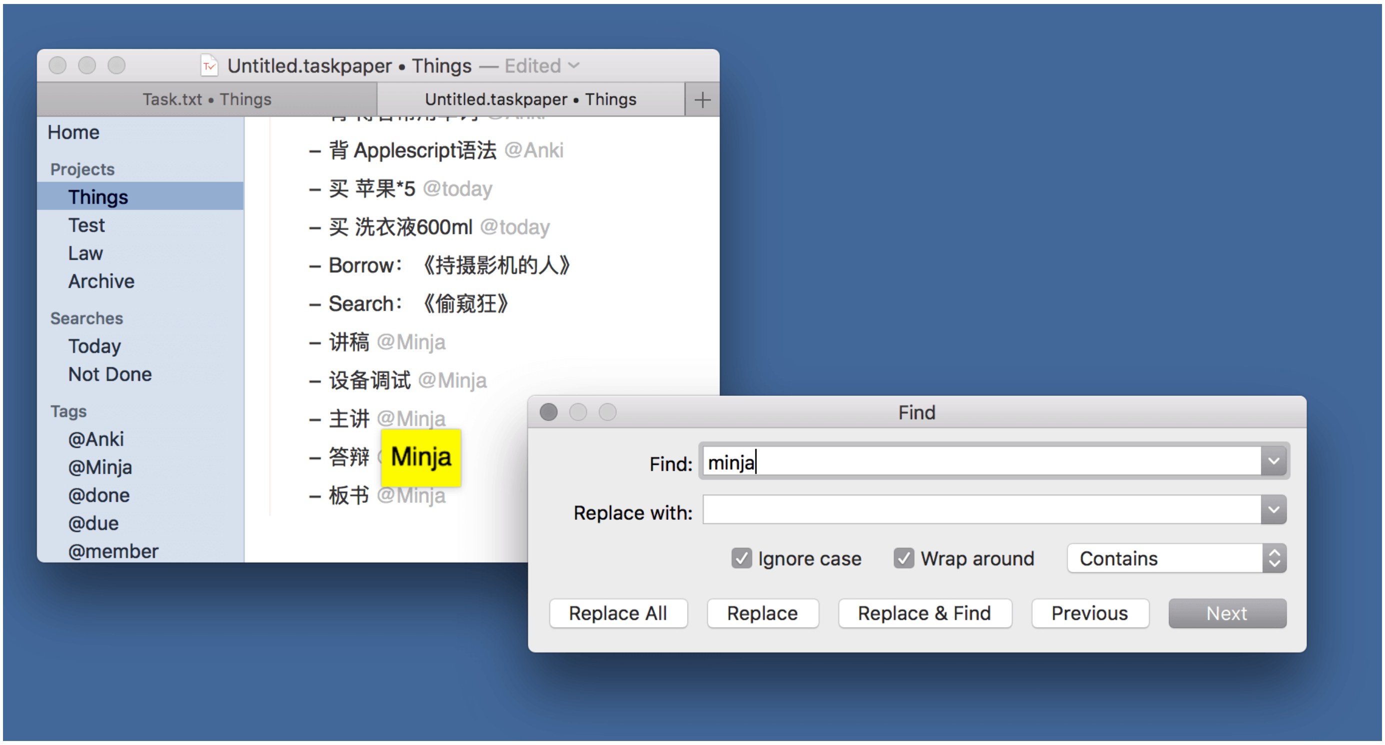
我使用较为频繁的有 @Anki() 标签组，括号里填入不同的学科（课目）。Anki 是一个记忆辅助 App，我拿它来背日语单词、法条和算法例子；它自带的提醒食之无味，我干脆就把 Daily 学习计划整合进了 TaskPaper。一览当天学习目标的时候，我只需要输入 @Anki，所有的课目都会出现；开始学习时，则在 searchbar 1 里键入@Anki=刑侦学这样的语句，只显示目前要背的几章。

@Anki 标签

搜索（过滤）语法

TaskPaper 支持两种模式的搜索：普通的搜索和 searchbar，但是只有后一种是我们要重点关注的过滤。像大多数编辑

器、任务管理工具一样，在 TaskPaper 里按下 ⌘command+F 快捷键可以打开一个搜索界面，但是这个普通搜索只能进行「找」和「替换」操作，在任务列表中高亮显示关键词。如其名字所示，不过是「Find」。这和其他文本编辑器一样是个规规矩矩的基础功能，只在于批量修改文字的时候有用。


Untitled.taskpaper • Things — Edited
Task.txt • Things
Untitled.taskpaper • Things
Home
Projects
Things
Test
Law
Archive
Searches
Today
Not Done
Tags
@Anki
@Minja
@done
@due
@member
Find
Replace All
Replace
Replace & Find
Previous
Next

- 背 Applescript语法 @Anki
- 买 苹果*5 @today
- 买 洗衣液600ml @today
- Borrow：《持摄影机的人》
- Search：《偷窥狂》
- 讲稿 @Minja
- 设备调试 @Minja
- 主讲 @Minja
- 答辩 @Minja
- 板书 @Minja

Find: minja
Replace with:
☑ Ignore case ☑ Wrap around Contains

普通搜索

对于任务管理，我们更需要的搜索其实是「过滤」，不仅仅是找到，还要把符合条件的任务汇集、展示给我们。让我们打开 searchbar，进行更灵活的搜索——过滤。过滤语法一眼看去很复杂：

@<属性><关系><修饰词><关键词>

例如：

@due<=[d]next Monday

先放松，不用每次都输入这几项，有时缺少一部分是没事的——只不过越详细越精确——你也可以直接输入关键词而不管任何语法。另外，再复杂的语法都可以保存下来，方便下次直接取用。

来看看语法中各个部分代表了什么。

关键词

关键词是一句过滤语句中最主要的的部分。其实一些简单的搜索，比如临时想找一下带有「庄严」的任务，直接搜「庄严」就可以。

但你也发现了问题——如果有一个项目是 投好庄严一票:（显然不是我们需要的），它也会出现在搜索结果里，但是我们**只要任务，不要项目**。此时单独的关键词就捉襟见肘，需要依靠其他部分打配合。

属性

接上。

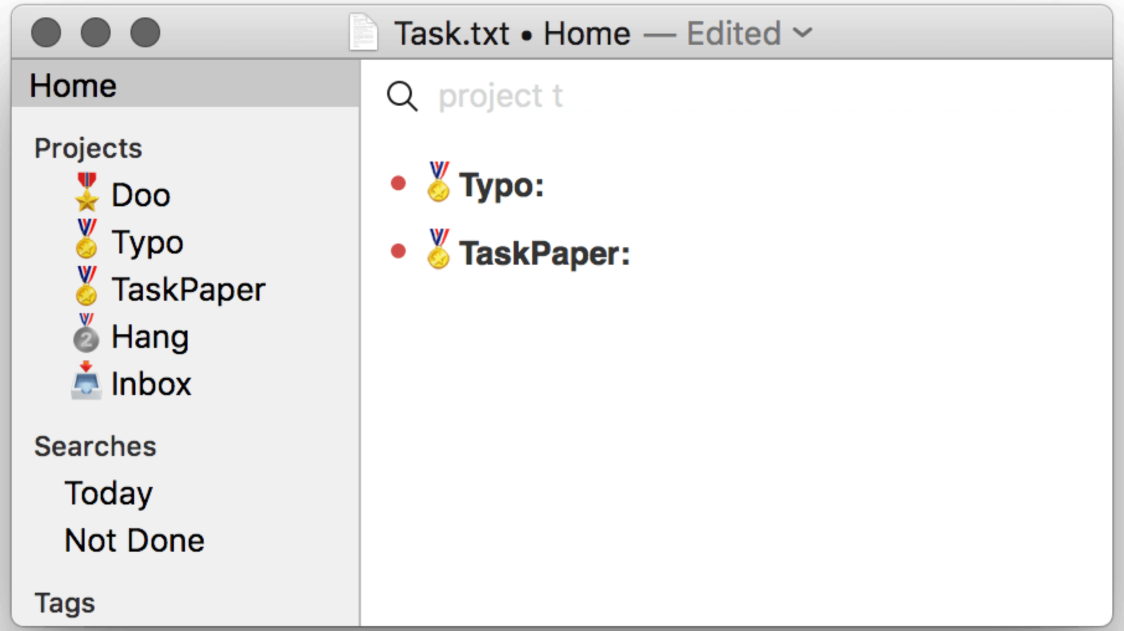
为了不让好庄严先生抢镜，可以用这个语句 @type = task 庄严，它的意思是只搜索带有「庄严」字样的任务。其中的 @type 就是一种属性，它还可以把过滤范围限定在项目或备注。TaskPaper 的属性如下表，一般记住 @type 系列就足够了。

- @type：匹配类型，有 project、task 和 note
- @text：匹配一行字符，这是过滤的默认属性，平时无需手动输入
- @id：匹配特殊的 ID（一般用不到）

如你所见，输入量未免大了点，我也不喜欢那么长一串。TaskPaper 为部分 @type 属性设置了缩写格式：

- @project = 你的项目名字 可以写成 project 你的项目名字
- @task = 你的任务名 可以写成 task 你的任务名
- @note = 你的备注 可以写成 note 你的备注

是不是就简单多了:


Task.txt • Home — Edited
Home
Projects
Doo
Typo
TaskPaper
Hang
Inbox
Searches
Today
Not Done
Tags

project t

- **Typo:**
- **TaskPaper:**

属性缩写

不过，标签组的过滤语法比较特殊，它和属性很像，但是没有缩写形式。最基本的格式是 @标签组名 = 标签备注。

关系

除了简单的 =，属性和关键词之间、标签组名和标签备注之间，还有逻辑关系、大小关系、字符匹配等几类。假设标签 @p(优先级数) 表示优先级高低，那么要找优先级大于 2 的任务，就可以用大于号：@p > 1。

各种完整的关系如下：

逻辑符号	意义	比较符号	意义
=	等于	<	小于

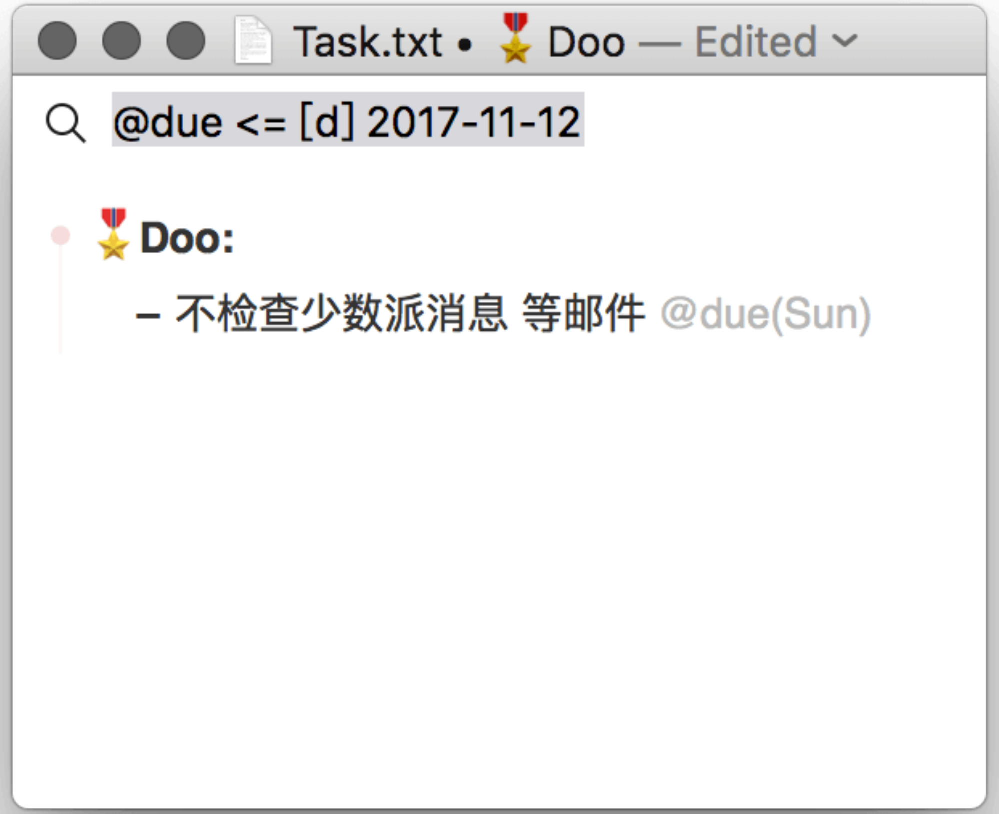
!=	不等于	>	大于
		<=	小于或等于
		>=	大于或等于
字符			**意义**
contains			包含
beginswith			以某字符开头
endswith			以某字符结尾
matches			包含（包括该行的标签）

最实用的，还是等于、不等于、大于、小于这类简单的符号。留心字符相关的 relation 符号，它们是第三人称单数格式，特别是 matches 以 es 结尾，容易疏忽。

修饰词

在特殊的情况下，为了让搜索结果更精确，我们还需掌握最后一个知识：修饰词。它们是一群包装在方括号里的英文字母，为过滤再加一层滤纸。用得最频繁的要属 [d]，表示**比较日期**。大量以日期为条件进行的过滤都需要它。

例如想过滤截止日期在 11 月 12 号之前的任务，就可以试试这个：


Task.txt • 🎖 Doo — Edited

@due <= [d] 2017-11-12

🎖Doo:

- 不检查少数派消息 等邮件 @due(Sun)

修饰词

TaskPaper 支持的所有修饰词如下表：

符号	意义
i	忽略大小写（默认）
s	不忽略大小写
n	比较数值
d	比较日期

组合使用

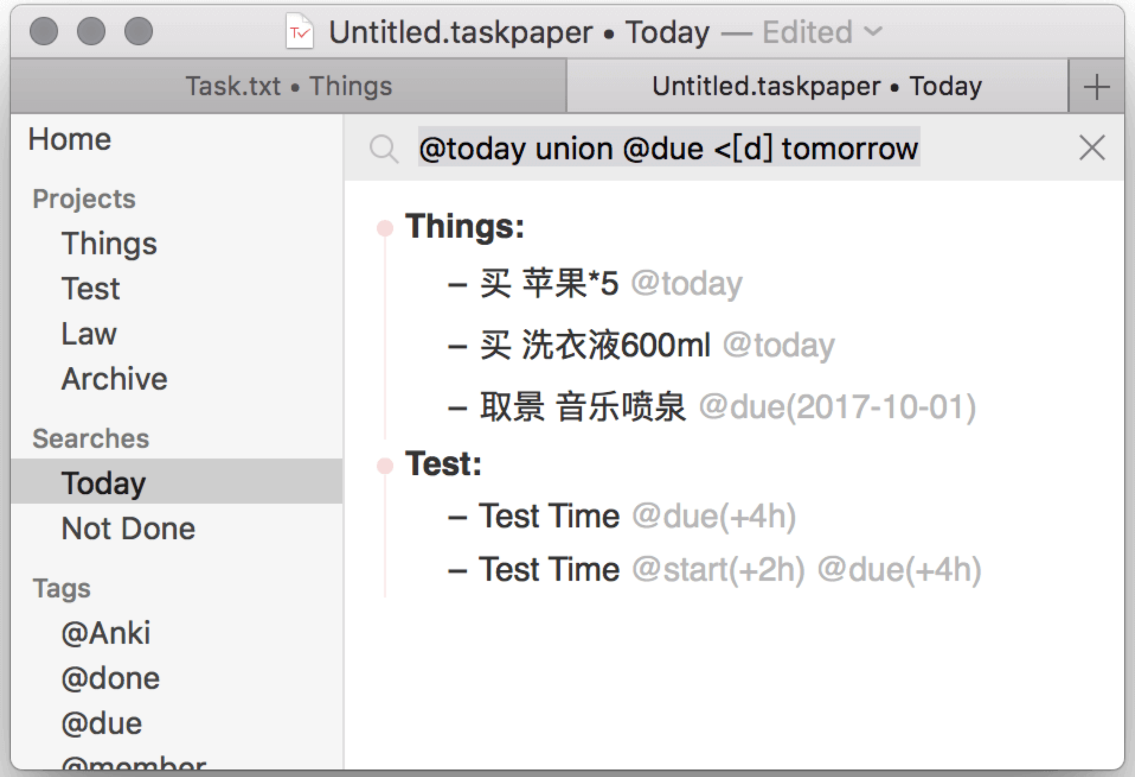
很多情况下我们要搜索的东西一句话描述不完，往往需要几句话组合在一起才能奏效。我们可以用一些词语把多个搜索语句连接起来：

符号	意义
union	或，满足一个即可
intersect	且，要同时满足两个
except	排除某个条件

特别需要留意的是，没有~~「A and B」~~这样的语法，A union B 才是满足 A 或 B 条件任务的合集。还是预设搜索动作里那个经典的例子：

@today union @due <[d] tomorrow

前半部分匹配你手动打上「今天」标签的行（包括项目和任务），后面则是绝对时间符合「今天」的行。组合在一起，才是完整意义上的「今日任务」。


Untitled.taskpaper • Today — Edited
Task.txt • Things
Untitled.taskpaper • Today
Home
Projects
Things
Test
Law
Archive
Searches
Today
Not Done
Tags
@Anki
@done
@due

@today union @due <[d] tomorrow

Things:
- 买 苹果*5 @today
- 买 洗衣液600ml @today
- 取景 音乐喷泉 @due(2017-10-01)

Test:
- Test Time @due(+4h)
- Test Time @start(+2h) @due(+4h)

Today searches

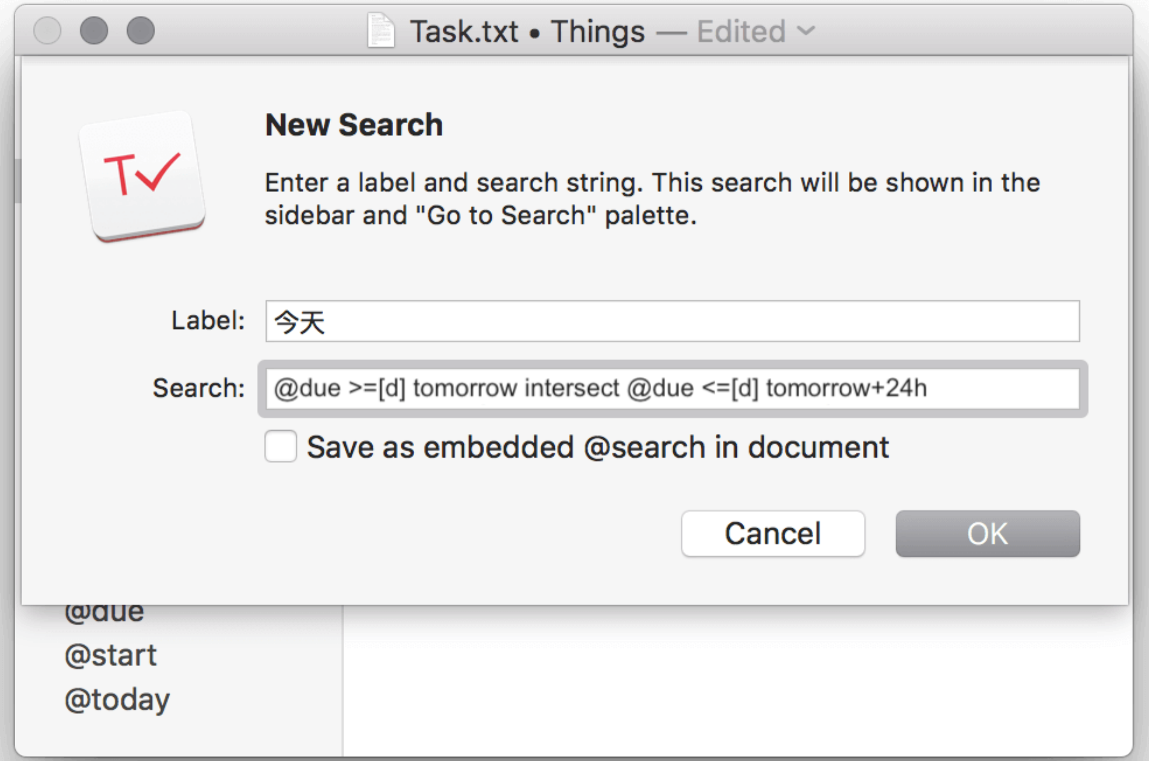
其他的组合搜索，也如法炮制。

searches 搜索动作

用户不禁要抱怨：每次都得输入一串字符，很难记住，输入量也不轻。TaskPaper 考虑到了这一窘况，在 sidebar 里增加了 searches 栏目，用于保存常用的搜索动作。预设的两个 searches 是 Today 和 Not Done，点一下即可。如本章开头介绍的，searches 和 Go To Anything 里都可以选中一个搜索名进行搜索。

你也可以自定义这一部分，把常用的搜索动作保存在里面。有两个方法可以做到这一点。最简单的是点击「菜单栏 - View - New Sidebar Search」，在「Label」内填入喜欢的搜索名，「Search」里则是准确的搜索条件。

例如我不满足 TaskPaper 预设时间过滤，需要添加一个过滤明天任务的 searches：@due >=[d] tomorrow intersect @due <=[d] tomorrow+24h，它表示过滤从今天午夜开始到后天凌晨之间的任务；再为这个 searches 取个名字「Tomorrow」，以上方代码框内的格式添进配置里。以此类推，再往里面增加注诸如「A head of time（提前完成）」、「This Week」之类的 searches。已经添加到 sidebar 内的 searches 也可以右键之进行修改。



New Search

Enter a label and search string. This search will be shown in the sidebar and "Go to Search" palette.

Label: 今天

Search: @due >=[d] tomorrow intersect @due <=[d] tomorrow+24h

☐ Save as embedded @search in document

如果你有了比较系统化的管理方式，需要批量添加一系列的搜索条件，可以直接修改 ~/Library/Application Support/TaskPaper/Configurations/searches.taskpaper 目录下的 searches 配置文件，在最后一行下面添加自己的 searches：

通过直接修改配置文件的方法，你可以同时录入多条 searches。每输入一条请换行、按 tab 缩进。

Searches 配置文件

补充包：习惯上的注意

TaskPaper 的标签灵活好用，但是不够「好看」，存在着三个「瑕疵」。

首先，TaskPaper 的标签只管自己所在的那一行，项目和母任务的标签不会影响到其下面的任务，项目和任务亲如父子，谈到标签却只有「各扫门前雪」。其实，这样的好处是可以减少过滤结果，只显示匹配的那一行，而不会把它下面附带的任务和子任务一股脑倒出来。如果你真的需要批量添加标签，尤其是给项目里的所有任务都贴上标签，可以全选它们，使用「cmd+t」进行打标签操作。

然后是标签组，TaskPaper 重视了搜索，却在一定程度上忽视了显示。sidebar 里只有标签组名而没有完整的 @tag(备注) 标签名，不可以像项目-子项目那样以层级的显示。作为全键盘用户我可以理解这种设计，因为使用 searchbar 搜索

的时候是不需要点按 sidebar 的，没有必要用一层层的标签占满侧边栏。所以，如果你很依赖有层级的标签，还是需要多多掌握搜索语法。

最后是有效标签的显示，这算是 TaskPaper 的不足之处。我有七八个项目，但一次只会查看一个（往往是「当日任务」这种），但是所有标签都会出现在 sidebar 和 Go To Anything 建议里；如果我把所有项目公私不分地堆在一个 taskpaper 文件里，很可能在检视课程和工作目标的时候，看见了来自「古都最好吃 100 家夜宵」的美食标签。**解决方法很简单，如果标签实在太多，或者把几个不同项目的标签堆在一起显得很违和，可以把项目分储在几个不同的 taskpaper 文件里。**同样是非常依赖的标签的 Thing 3，在处理有效标签的时候就比较讨巧，只显示当前所查看项目具有的标签，显的清爽一些。

但是仔细想想，在这些略微的不便背后，其实正是 TaskPaper 定位和搜索的哲学：不必多想。表现在 sidebar 上，就是用不着一层一层往下翻找，简单的（现成的）瞥一眼直接点，没有就打开 searchbar 直接搜。

> Don’t make me think.

TaskPaper 就是那么简单直接。

用标签组归类不确定的任务

做任务管理，重要的目的就是进行可行性判断，任务的前提

事实上非常关键，有时它直接决定了是否执行一件事。任务管理的直接作用，就是做判断、做规划，哪些任务能做、何时何地能做。

其实 GTD 里的执行也不是真正去做事情，而是做之前那一刻再做一次可行性判断：

> 每一个行动决策都是凭直觉作出的。问题在于，如何从希望这个选择是正确的过渡到对它的正确性深信不疑。——David Allen《Getting Things Done》

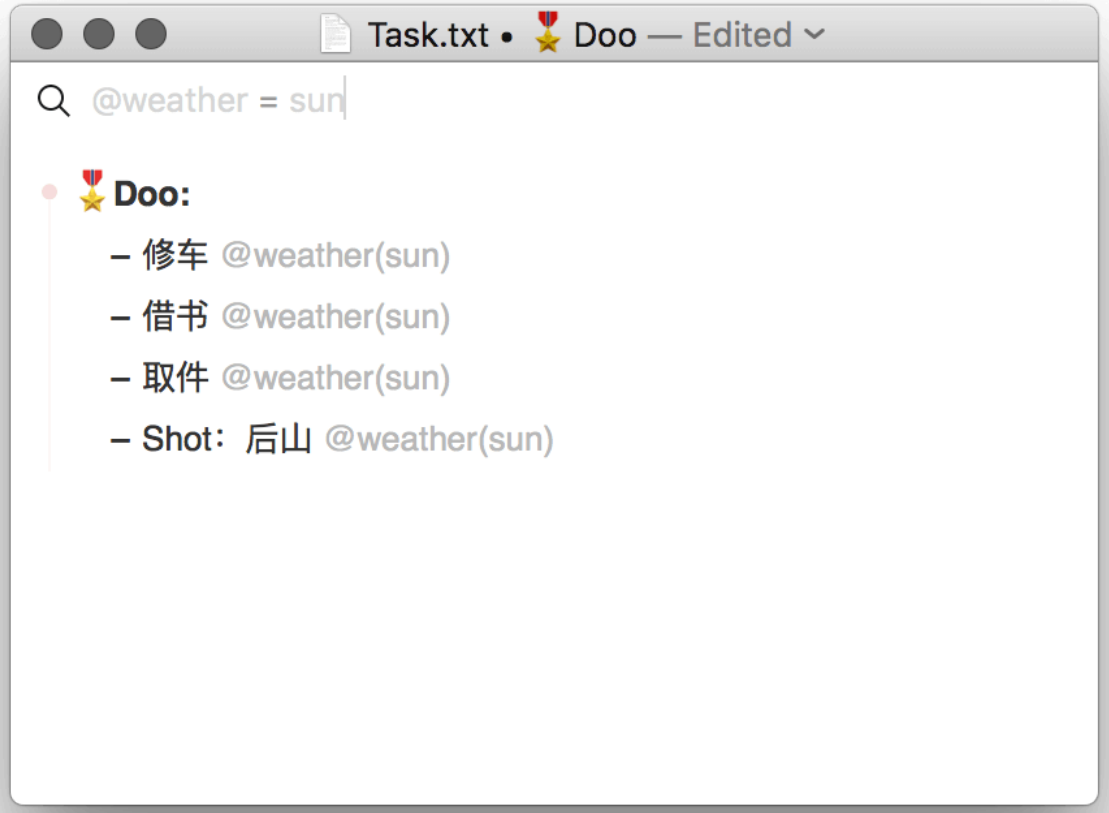
任务管理帮助我们做出合理的判断。从做判断的角度看，我们给任务打上的标签，相当一部分是虚拟时态，是一个假设，比如 @Chap 就可以理解为「如果 Chap 先生在，这个任务才能执行」。

以典型的影响因素——天气为例，我们常常会看着天空，想：啊，明天天晴的话就去超市买餐巾纸。注意，此时你不确定这个任务要不要做，这类尚不确定的任务不记下来就容易流走，但是一股脑塞进 Task Paper 则会让人无列表变得复杂无比。

用单独的一个标签组归类它们吧。就拿天气来说，我喜欢使用这组标签 @weather()，括弧内填入不同的天气。在我写这一章的当天，车子不幸爆胎，在雨夜里推车踽踽独行的我记下了一条任务：

大意就是明天天晴的话，就去修车子。光从这一条来看，似乎设置条件的意义不大；但是有个八九条不确定任务时，就

很有必要了：**两种天气下的任务，一旦各自积累到一定数量，就代表了两种截然不同的日程安排。**



@weather = sun

🎖 **Doo:**
- 修车 @weather(sun)
- 借书 @weather(sun)
- 取件 @weather(sun)
- Shot：后山 @weather(sun)

条件任务

生活在天气变化无常的亚热带，下雨和晴天基本决定了出门的频率。比如九月底的某一天，当地连续下了五日雨，我攒了一堆要外出处理的事情；运气好能够放晴的话，我就去图书馆借《睡鼠说》（没有合适的电子版）、去自提柜取邮包，再把前几日拖欠的短视频取景任务完成了；如果还是下雨，就待在屋子里，烘烘衣服、补补美剧。录入这些任务的时候是不需要考虑其可行性、先后顺序的，因为最大的干扰

因素「天气」尚不清楚——**对于不确定的任务，务必记下，日后过滤。**事先已经把各种天气下的任务都考虑进去，就不太会遗漏。

早起之后我看一眼天气，就依据真实情况，过滤出当日最适合执行的任务。

读者想必会问：为什么不直接使用 sun 和 rain，那样不是可以节约输入时间吗？此时的一点小便宜，在过滤的时候就要麻烦你输入更多字符了。何况除了晴雨，我也会遭遇其他情况，如果把每种天气都直接设为标签，即使全屏显示 TaskPaper 窗口，sidebar 也不够用。此外，依靠自动补全，设置标签组也花不了多大力气。

补充包：屡试不爽的精力标签组

除了天气，还会有其他的影响因素——精神状态就是其中一种。我们很容易忽视精力对人物的影响，其实人非铁打，总是有一个精力的峰谷期，需要合理根据精力规划任务。精力标签是一个显眼的提示方式，@Hum 在《用更现代的方式做任务管理》一书中，详细讲解了他如何用 @EasyDo 标签标出精神低迷时可以做的事情。我则把回复来信、清空消息提醒这类**不紧急、不费力的事情打上 @dead 标签**，安排在容易打瞌睡的午后来做。

而借助十一月上旬更新的 workflow，我还制作了一个提醒我注意精力分配的小工具。我个人有个不算太好的习惯，即容易工作、学习到「忘我」，其实在精力不佳的时候还不停下

休息，对于身体和效率都有害。为了调整自己的作息，我借助 Workflow 来分析健康状况并给出精力标签建议。比如说我熬夜而致使手机所记录的睡眠时间减少，次日在手机上添加任务时，workflow 就会发现我没有好好睡觉，随即弹出温馨提示「还要继续工作吗」，此时我依然可以正常输入任务，但是每条任务后面就会打上 @dead 的标签，这就提示我要量力而行，尽量安排不烧脑的事情。

Workflow 精力标签

分析睡眠时间只是窥豹一斑，运动、进食都可作为精力标签的参考依据。欢迎下载我这个 Workflow，修改其中的健康数据类型和标签，以符合你的使用习惯。

其实精力标签有一整组，下面是我所理出的一个完整表格。显然，让它们都挤进列表反而会失去指示作用，建议你像我一样，依据自己精力的「峰期」和「谷期」来选一两个使用。

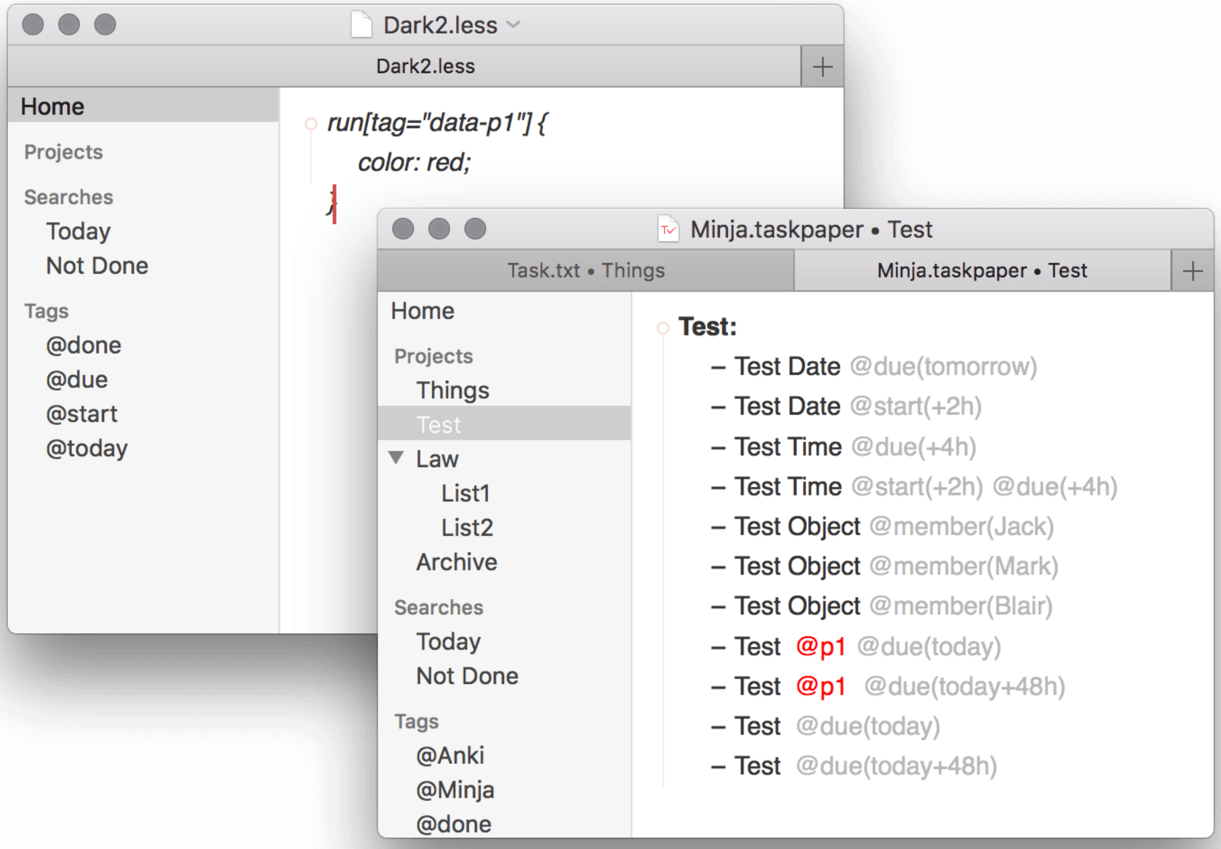
标签	所需精力	例子
@focus	耗时费力	写这篇文章 ;)
@long	耗事不费力	看美剧、听播客
@quick	省时不费力	烧开水、泡茶
@dead	精神低迷时也能做	整理桌面
@every	周期性但不费力	买水果

优先级

iPhone 自带的提醒事项可以设置优先级，我过去热衷于画上一个个红色惊叹号，仿佛自己很「商务」，很「有效率」。现在看来这有些自欺欺人，仿佛得了甲亢。但不可否认，优先级是一种醒目的提示。

TaskPaper 没有自带优先级，但它的标签系统能够实现。一个操作简单、效果直观的方法是使用 emoji 表情 ‼️。**单独的 emoji 是不能被搜索到的**，我们需要把它作为优先级属性的具体值，形成这样的一系列标签：

如果你对视觉样式的要求比较高，觉得打括号不过雅致，可以回忆一下 StyleSheet 的基本语法。只需要把@p1这个标签设成红色就解决问题：


Dark2.less
Dark2.less
Home
Projects
Searches
Today
Not Done
Tags
@done
@due
@start
@today
Minja.taskpaper • Test
Task.txt • Things
Minja.taskpaper • Test
Home
Projects
Things
Test
Law
List1
List2
Archive
Searches
Today
Not Done
Tags
@Anki
@Minja
@done

```
run[tag="data-p1"] {
    color: red;
}
```

```
Test:
    – Test Date @due(tomorrow)
    – Test Date @start(+2h)
    – Test Time @due(+4h)
    – Test Time @start(+2h) @due(+4h)
    – Test Object @member(Jack)
    – Test Object @member(Mark)
    – Test Object @member(Blair)
    – Test @p1 @due(today)
    – Test @p1 @due(today+48h)
    – Test @due(today)
    – Test @due(today+48h)
```

优先级

优先级标签会乖乖出现在 sidebar 里，可直接作为一个过滤条件使用。我坚持「今日事，今日毕」的原则，一般只使用最高一级的优先级标签，不喜欢次级的任务也来抢镜头；而你可能更希望列表整整齐齐，任务呈一二三梯队排好，你我的不同需求这一套标签都能满足。

TaskPaper 不能直接搜索 emoji 的确可惜，第一次手动输入稍微麻烦了些，之后依靠自动补全能省力点。

四象限

在优先级的基础上，再引入一组标签，就可以用 TaskPaper 实现四象限法，对任务进行先后轻重的分类。

语法	作用
重要，紧急	@p intersect @due<=today+24h intersect @due>=today
重要，不紧急	@p intersect @due<=today+24h
不重要，紧急	except @p intersect @due<=today+24h intersect @due>=today
不重要，不紧急	except @p intersect @due>=today+24h

我认为的「紧急」就是「今天就得完成的任务」，你可能有不同理解，可以根据自己的需要修改这个表格。四象限法的搜索语句无非四句话，稳定不变，非常适合存到 searchbar

里。